

PREVIOUS SUBMISSION

Applicants respectfully request that the Examiner not enter the Amendment submitted January 3, 2006 and responsive to the Office Action mailed November 1, 2006. In its place, Applicants respectfully request that the Examiner instead consider Applicants amendments, remarks and the newly presented claims provided herein.

REMARKS

Claims 1-38 were rejected by the Examiner. Applicants added Claims 39-53. Claims 1-53 are still pending. Claims 1, 15, 19, 25, 27, 29, and 36 have been amended. Reconsideration is respectfully requested in view of the amendments above and the following remarks.

Claim Rejections under 35 U.S.C. § 103(a)

Claims 1, 3-27, 29-34, and 36-38 were rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,857,201 issued to Wright et al. (hereinafter referred to as "Wright") and further in view of U.S. Patent No. 5,295,222 issued to Wadhwa et al. (hereinafter referred to as "Wadhwa").

Claim 2 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Wright as applied to claim 1 above, and further in view of U.S. Patent No. 6,880,126 issued to Bahrs et al.

Claim 28 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Wright as applied to claim 27, and further in view of U.S. Patent No. 6,754,670 to Lindsay et al.

Claim 35 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Wright as applied to claim 29 above, and further in view of U.S. Patent No. 5,604,906 issued to Murphy et al. Applicants respectfully traverse.

ARGUMENT

I. Applicants respectfully assert that the 35 U.S.C. § 103 rejection is improper on the grounds both that, contrary to the Examiner's arguments, there is no motivation to

combine the teachings of Wright with those of Wadhwa and that Wright teaches away from its combination with Wadhwa.

The Examiner, citing Wadhwa 7:16-18, stated that a person of ordinary skill in the art would have been motivated to provide Wright with a data model that can be readily re-used in subsequent applications. (Office Action mailed 11.01.2005; Advisory Action mailed 02.01.2006.) Explaining this motivation, the Examiner states that “passage suggests that entity relationship models enable subsequent reuse. Reuse is a basic objective of object oriented software, and allows a single implementation to be reused by a variety of applications, or in the case of Wright, a variety of clients, in an effort to reduce implementation costs. Therefore, Wadhwa provides motivation” for its combination with Wright. (See Advisory Action mailed 02.01.2006.)

In view of the arguments set forth below, Applicants respectfully assert that the Examiner’s arguments are misplaced on at least the grounds that the Examiner’s cited motivation for combining Wadhwa with Wright – “re-use” – exists in Wright and, therefore, no motivation as asserted by the Examiner may be found for their combination. In addition, the teachings of Wright contradict the teachings of Wadhwa thereby prohibiting their combination as a basis for a valid rejection under 35 U.S.C. § 103.

- a. Wright teaches away from Wadhwa in that Wright teaches platform independent applications development and coding whereas Wadhwa is directed to resolving issues surround the development of code targeted to many specific hardware platforms.**

If a first prior art reference teaches away from a second prior art reference, that finding alone can defeat an obviousness claim based on a combination of the two references. *See, Winner Int’l Royalty Corp v. Wang*, 202 F.3d 1340 (Fed. Cir. 2000). A reference may be said to teach away when a person of ordinary skill, upon reading the first reference would be led in a direction divergent from the path taken in the second reference. *See, In re Haruna*, 249 F.3d 1327 (Fed. Cir. 2001).

Wright discloses a FormLogic engine allowing applications to execute on a variety of platforms. (See, e.g., Wright at 5:16-29.) As expressly stated by Wright, the FormLogic engine, the engine upon which the applications of Wright are designed to operate, “is a hardware independent

virtual machine that allows a single application to work on various hardware platforms.” (Wright at 5:25-26.)

Applicants agree with the Examiner, Wadhwa generally teaches object oriented programming. Specifically, Wadhwa discloses computer-aided software engineering facilities for assisting programmers or system developers in the design, development and testing of computer programs for use in multiprocessor systems. (See, e.g., Wadhwa at 1:16-19.) According to Wadhwa, in order to “design an application using the CASE facility, a developer must decompose the application into specified logical parts, and assemble them into a program.” (Wadhwa at 6:39-42.) “The different parts of an application are expressed as entities and are linked by relationships.” (Wadhwa at 6:44-46.) “With the Entity-Relationship model in place program construction can begin.” (Wadhwa at 11:21-22.)

In Wadhwa, entity-relationship models are employed to simplify the generation of code for an application intended for use on variety of hardware platforms. (See, e.g., Wadhwa at 1:16-19.) In other words, Wadhwa describes a system wherein a model of a desired application is developed. Once an application is modeled, Wadhwa provides CASE facilities which are configured to generate code for each of the targeted hardware platforms on which the application is intended to execute. To be clear, Wadhwa is directed to providing, potentially, numerous iterations of code for a single application such that the application may be executed on a variety of hardware platforms. It is in this regard that Wadhwa teaches the use of a reusable model – for the sole purpose of generating, for a single application, numerous versions of code such that the single application may be executed on a variety of hardware platforms.

From the indisputable teachings of both Wright and Wadhwa, it is clear that the implementations in each are in conflict – Wadhwa teaching platform dependence and, in stark contrast, Wright teaching platform independence. On this basis alone, the combination of Wadhwa and Wright should be rejected as this basic distinction – platform independent versus platform dependent code development – would indisputably lead any person of ordinary skill in the art reading Wright down a path entirely “divergent from the path taken in Wadhwa”. *See, In re Haruna.*

In fact, Wright should be considered to entirely supplant the teachings of Wadhwa. Based on the express teachings of Wright, it would be evident to one of ordinary skill in the art that to deploy an application in the Wright model one would need to develop and code only one version of the desired application. Thus, Wright clearly obviates any need for the entity-relationship model of Wadhwa through the use of Wright's "hardware independent virtual machine that allows a single application to work on various hardware platforms". Again, a person of ordinary skill in the art reading Wright would be led on a path – platform independent code generation – entirely divergent from the path taught by Wadhwa – platform dependent code generation. *Id.*

Accordingly, Applicants respectfully assert that the teachings of Wright are in conflict with the teachings of Wadhwa and, further, the Wright disclosure teaches away from its combination with Wadhwa. There is no suggestion to combine if a first reference teaches away from its combination with a second reference. *See, Tec Air, Inc. v. Denso Mfg. Michigan Inc.*, 192 F.3d 1353 (Fed. Cir. 1999). Therefore, Applicants respectfully request that the Examiner reconsider the combination of the Wright and Wadhwa references, withdraw the rejections based on their combination and allow Claims 1, 3-27, 29-34, and 36-38 rejected thereunder.

- b. The Examiner's cited motivation for the combination of Wadhwa with Wright – the motivation of "re-use" – should be rejected as Wright expressly teaches re-use in the form "a hardware independent virtual machine that allows a single application to work on various hardware platforms."**

A showing of obviousness requires a motivation or suggestion to combine or modify prior art references, coupled with a reasonable expectation of success. *See, Boehringer Ingelheim Vetmedica, Inc. v. Schering Plough Corp.*, 320 F.3d 1339 (Fed. Cir. 2003). A showing of a suggestion, teaching, or motivation to combine must be clear and particular; broad conclusory statements regarding the teaching of multiple references, standing alone, are not evidence. *See, In re Dembiczak*, 175 F.3d 994 (Fed. Cir. 1999). The test for establishing an implicit teaching, motivation, or suggestion is what the combination of statements of one or more prior art references would have suggested to one of ordinary skill in the art. *See, In re Kotzab*, 208 F.3d 1352 (Fed. Cir. 2000). Importantly, these statements must be considered in the context of the teachings of the entire

reference. *Id.* There is no suggestion to combine if a first reference teaches away from its combination with a second reference. *See, Tec Air, Inc. v. Denso Mfg. Michigan Inc..*

In addition to those reasons stated above and believed to justify the withdrawal of the claim rejections based on the combination of Wright and Wadhwa, Applicants respectfully assert that the rejections based on the combination of Wright and Wadhwa should also be withdrawn on the grounds that there is no motivation for their combination for at least the following reasons.

Applicants and the Examiner agree, "Wright does not expressly teach a data model". (Office Action mailed 11.01.2005, p. 6.) As such, Applicants respectfully assert that there can be no motivation for one of ordinary skill in the art merely to replace the implied model of Wright with the "model" of Wadhwa. Therefore, in order to properly combine the teachings of the Wright and Wadhwa references, there must be some alternative motivation. Thus, according to the Examiner and as mentioned above, the impetus for the combination of the Wright and Wadhwa references purportedly lies in the motivation of one of ordinary skill in the art to provide Wright with "a data model that can be readily re-used in subsequent applications."

As stated above, Wright discloses a FormLogic engine allowing applications to execute on a variety of platforms. (See, e.g., Wright at 5:16-29.) As expressly stated by Wright, the FormLogic engine, the engine upon which the applications of Wright are designed to operate, "is a hardware independent virtual machine that allows a single application to work on various hardware platforms." (Wright at 5:25-26.) With this statement, Wright unquestionably contemplates and provides for the extensive reusability of applications developed and coded for use therewith. Therefore, a person of ordinary skill in the art reading Wright **would not** be motivated to provide Wright "with a data model that can be readily re-used in subsequent applications" because one of ordinary skill in the art reading Wright would easily and immediately recognize that Wright already provides for the reusability of applications through Wright's express teaching of the use of platform independent application development and coding tools.

In view of the above arguments, Wright clearly teaches away from its combination with Wadhwa – Wright seeks to develop and code a single application while Wadhwa seeks to generate

application code for each and every different hardware platform upon which the desired application is intended to execute. Thus, as Wright already teaches reusability, one of ordinary skill in the art would not be motivated on those same grounds to seek out utilize the incompatible teachings of Wadhwa.

Accordingly, Applicants respectfully assert that the teachings of Wright are in conflict with the teachings of Wadhwa and, further, the Wright disclosure teaches away from its combination with Wadhwa. There is no suggestion to combine if a first reference teaches away from its combination with a second reference. *See, Tec Air, Inc. v. Denso Mfg. Michigan Inc.*, 192 F.3d 1353 (Fed. Cir. 1999). Therefore, Applicants respectfully request that the Examiner reconsider the combination of the Wright and Wadhwa references, withdraw the rejections based on their combination and allow Claims 1, 3-27, 29-34, and 36-38 rejected thereunder.

c. The Examiner's cited motivation for the combination of Wadhwa with Wright – "to reduce implementation costs" – should be rejected on the basis that such an assertion is not grounded in reality and on the basis that Wadhwa expressly acknowledges the HIGH COSTS of its methodology.

Further, the Examiner stated that a person of ordinary skill in the art would be motivated to combine the teachings of Wadhwa with Wright "in an effort to reduce implementation costs". In both Wright and Wadhwa, a programmer will have to develop an application for coding. In both Wright and Wadhwa, a programmer will have to code the developed program for execution. Where Wright and Wadhwa diverge, is that Wright requires the application to be generated once whereas Wadhwa requires that the application be generated over and over again until appropriate code for each hardware platform on which the application is to be run has been created. In at least this regard, it is Wright, not Wadhwa, which provides the greatest reduction in implementation costs. Someone familiar with Wright would not be motivated to add Wadhwa teachings because that would be a step backwards.

Alterations desired or needing to be made to the developed applications will be more efficiently made under the Wright implementation than under the Wadhwa implementation. Specifically, under Wadhwa, for a change to be made to an existing application, the entity-relationship model will first have to be changed. Following entity-relationship model modifications,

the code for each hardware platform will again have to be generated and redistributed according to specific hardware and associated coding limitations. Wadhwa expressly acknowledges the HIGH COST of its methodology stating that “small changes in this system can have large consequences. In general, any entity type that owns, uses or includes entities that have been changed will have to be reprepared or modified.” (17:1-4).

In contrast, under the Wright implementation, the application to be changed need only be modified as desired and then redistributed to its targeted devices. As such, Wright clearly provides the greatest reduction in implementation costs in both origination of application code as well as application code modification. In other words, Wright’s platform independent, single iteration application development and coding teachings provide greater reductions in implementation costs than will the repetitive code generation teachings of Wadhwa – without which the entity-relationship model of Wadhwa no longer provides reusability.

Accordingly, Applicants respectfully assert that the Examiner’s cited motivation for combining the Wright and Wadhwa references is misplaced based on at least the reasons provided herein. Applicants respectfully request that the Examiner reconsider the combination of the Wright and Wadhwa references, withdraw the rejections based on their combination and allow Claims 1, 3-27, 29-34, and 36-38 rejected thereunder.

II. Even if inappropriately permitted, the combination of Wright and Wadhwa fails to disclose, teach or otherwise suggest a data model as claimed by Applicants.

- a. The Examiner and Applicants agree, Wright DOES NOT “expressly disclose a data model defining one or more data element, data relationship, data dependency and data distribution attributes”.**

The Examiner and Applicants agree, Wright “does not expressly disclose a data model defining one or more data element, data relationship, data dependency and data distribution attributes”. (Office Action mailed 11.01.2005, p. 6). In contrast, Wright et al. detail an application based, programmatic solution directed to improving the capability of custom designed programs.

- b. Wadhwa, through its usage of an “entity-relationship model”, discloses a model representing only the association (relationship) of data (entities).**

The Examiner has stated that “Wadhwa teaches that a data model that defines at least data relationships can be used for generation and distribution of applications.” (Office Action mailed 11.01.2005, p. 6). Nowhere in Wadhwa is an entity-relationship model defined to include “data dependency and data distribution attributes” as claimed by Applicants.

Specifically, Wadhwa, in its description of a model used to generate platform dependent application code, states that the “different parts of an application are expressed as entities and are linked by relationships.” (6:44-46). Continuing, Wadhwa states that “an entity is something real or abstract about which information is recorded.” (6:47-48). A relationship, according to Wadhwa, is an “association between entities” and is “defined by attributes”. (6:59-63). Wadhwa never mentions a data model defining data dependencies or data distribution attributes in its discussion of an entity-relationship model – as will be noted in greater detail below, Wadhwa has absolutely no use for such attributes. Nowhere does Wadhwa discuss data dependency or data distribution attributes, let alone a data model defining data element, data relationship, data dependency and data distribution attributes required for interfacing a mobile software application with a backend application.

c. Directed to hardware platform specific application code generation in a multiprocessing or distributed processing environment, there is no basis in Wadhwa for the use of data distribution and data dependency attributes.

- i. The present application contemplates asynchronous communication between periodically connected mobile computing systems and devices whereas Wadhwa contemplates persistently connected processing nodes.

In the present application, a data model defining data element, data relationship, data dependency and data distribution attributes required for interfacing a mobile software application with a backend application is provided. In this vein, the present application contemplates the need to share data by and between multiple disconnected clients as well as to communicate asynchronously between the periodically disconnected clients and one or more backend applications. In such an environment, managing data dependencies and distribution is critical.

As a solution for generating application code in a multiprocessor or distributed processing environment, Wadhwa contemplates an environment of persistent network connectivity –

multiprocessing and distributed processing systems typically being hardwired together over one or more network connections. As such, the system of Wadhwa may be considered a single entity consisting of each connected device participating in processing, thereby, all communications between the devices may be considered internal communication within the distributed processing configuration.

ii. Multiprocessor and distributed processing solutions utilize highly structured applications with extensive process order management.

As suggested by the manner in which the present application describes and claims its data model, the data model of the present application plays an important role in interfacing a mobile software application with a backend application. This too provides a context in which it can be seen that the data model of the present application is not disclosed, taught or otherwise suggested by Wadhwa.

In contrast to the environment in which the present application is designed to operate, Wadhwa is directed to operation in a highly structured and managed operating environment. By their very nature, distributed or multiprocessor applications – whose fundamental difference with conventional, single node applications is that processing operations are shared among nodes to increase processing power and decrease processing time – direct which portion of what processing needs to occur when, in accordance with what priority, which portions of processing may be performed in parallel, etc. Such an environment is distinct from a remote or mobile computing system at least in the aspect that random connections, concurrency, and other issues that accompany the remote computing system are absent.

iii. Wadhwa is concerned only with the distribution of software generated from the entity-relationship model, performed by a Software Distribution System independent of the entity-relationship model.

Wadhwa discloses a “Software Distribution System which automates and controls migration of an application. The system manages the release of software to targeted computers. The Software Distribution System solves the problem of synchronizing distribution of software located, for example, on hundreds of personal computers.” (6:24-29). As such, distribution in Wadhwa – that is

distribution related to the dissemination of a distributed or multiprocessor application to each of its targeted nodes – is orchestrated by a Software Distribution System entirely separate and apart from Wadhwa's "Entity-Relationship model". Consequently, Wadhwa lacks any motivation to incorporate attributes concerning "software distribution," let alone the data dependency and distribution attributes contemplated by the present application, into its purported data model. In fact, Wadhwa's specific inclusion of a Software Distribution System teaches away from the incorporation of any aspect of distribution into its Entity-Relationship model.

- d. In contrast to Wadhwa's Entity-Relationship model, the data model of the present application defines and is actively employed, accessed or otherwise leveraged in interfacing a mobile software application with a backend application.**

As described above, Wadhwa develops an Entity-Relationship model from which hardware environment or platform specific application code is generated. "The entity –relationship model is not the actual program." (7:24-25). Subsequent to code generation, the entity-relationship model from which application code was developed is stored for possible later re-use or otherwise discarded. (See, e.g., 11:59-60). "When the modules of an application have been successfully prepared, they are ready to be transported to their target environments," *i.e.*, the application modules and not the entity-relationship model are transported to their target environments. (15:42-44).

In contrast, the present application describes and claims "a data model defining data elements, data relationships, data dependencies and data distribution attributes required for and actively employed in interfacing a mobile software application with at least one of the plurality of backend applications". Unlike, Wadhwa, an embodiment of the present application anticipates leveraging the data model it provides during interfacing between a mobile software application and a backend application.

In view of at least the foregoing arguments, Applicants respectfully submit that the combination of Wright and Wadhwa, fails to teach, disclose or otherwise suggest a method comprising, among other elements, "a data modeling program allowing creation of a data model defining data elements, data relationships, data dependencies and data distribution attributes required for interfacing a mobile software application with at least one of the plurality of backend

applications” as claimed in Applicants’ Claim 1. Accordingly, Applicants respectfully request that the Examiner reconsider the rejection of Claim 1, withdraw the rejection and allow Claim 1.

Claims 2-14 depend from and provide further patentable limitations to Claim 1. Accordingly, Applicants respectfully request that the Examiner reconsider the rejection of Claims 2-14, withdraw the rejections and allow Claims 2-14.

In view of at least the foregoing arguments, Applicants respectfully submit that the combination of Wright and Wadhwa, fails to teach, disclose or otherwise suggest “a first data model defining data elements, data relationships, data dependencies and data distribution attributes required for interfacing a mobile software application with a backend application and referencing at least one enterprise object associated with the backend application” as claimed in Applicants’ Claim 15. Accordingly, Applicants respectfully request that the Examiner reconsider the rejection of Claim 15, withdraw the rejection and allow Claim 15.

Claims 16-25 depend from and provide further patentable limitations to Claim 15. Accordingly, Applicants respectfully request that the Examiner reconsider the rejection of Claims 16-25, withdraw the rejections and allow Claims 16-25.

In view of at least the foregoing arguments, Applicants respectfully submit that the combination of Wright and Wadhwa, fails to teach, disclose or otherwise suggest a method comprising, among other elements, “a software tool for creating a mobile data model, the mobile data model associated with data from the enterprise software system and defining data elements, data relationships, data dependencies and data distribution attributes required for interfacing a mobile software application with the enterprise software system” as claimed in Applicants’ Claim 26. Accordingly, Applicants respectfully request that the Examiner reconsider the rejection of Claim 26, withdraw the rejection and allow Claim 26.

Claims 27-28 depend from and provide further patentable limitations to Claim 26. Accordingly, Applicants respectfully request that the Examiner reconsider the rejection of Claims 27-28, withdraw the rejections and allow Claims 27-28.

In view of at least the foregoing arguments, Applicants respectfully submit that the combination of Wright and Wadhwa, fails to teach, disclose or otherwise suggest a method comprising, among other elements, "mobility deployment code for deploying at least a portion of a data model defining data elements, data relationships, data dependencies and data distribution attributes required for interfacing a mobile software application with an enterprise backend application or object to a mobile computing device" as claimed in Applicants' Claim 29. Accordingly, Applicants respectfully request that the Examiner reconsider the rejection of Claim 29, withdraw the rejection and allow Claim 29.

Claims 30-35 depend from and provide further patentable limitations to Claim 29. Accordingly, Applicants respectfully request that the Examiner reconsider the rejection of Claims 30-35, withdraw the rejections and allow Claims 30-35.

In view of at least the foregoing arguments, Applicants respectfully submit that the combination of Wright and Wadhwa, fails to teach, disclose or otherwise suggest "mobility deployment code for deploying at least a portion of a data model defining data elements, data relationships, data dependencies and data distribution attributes required for interfacing a mobile software application with an enterprise backend application or object to a mobile computing device" as claimed in Applicants' Claim 36. Accordingly, Applicants respectfully request that the Examiner reconsider the rejection of Claim 36, withdraw the rejection and allow Claim 36.

Claims 37-38 depend from and provide further patentable limitations to Claim 36. Accordingly, Applicants respectfully request that the Examiner reconsider the rejection of Claims 37-38, withdraw the rejections and allow Claims 37-38.

NEWLY PRESENTED CLAIMS

Applicants added Claims 39-44 above. Applicants represent that the addition of claims 39-44 adds no new subject matter to the present application. Further, Applicants represent that the newly presented claims are supported by the instant specification and the provisional patent application to which is claims priority and all associated Figures. Applicants respectfully request that the examiner consider the newly presented claims and allow Claims 39-44.

Application No. 09/848,952
Amendment dated May 1, 2006
Reply to Office Action of November 1, 2005

CONCLUSION

In light of the remarks set forth above, Applicants believe that they are entitled to a letters patent in the present matter. Applicants respectfully solicit the Examiner to expedite prosecution of this patent application to issuance. Should the Examiner have any questions or feel that further prosecution of this matter may be expedited through an interview, the Examiner is encouraged to telephone the undersigned.

The Commissioner is authorized to charge any additional fees which may be required, including petition fees and extension of time fees, to Deposit Account No. 23-2415 (Docket No. 26625-703).

Respectfully submitted,

Date: May 1, 2006

By: B. A. Dietzel
Brian A. Dietzel
Registration No. 44,656

WILSON SONSINI GOODRICH & ROSATI
650 Page Mill Road
Palo Alto, CA 94304-1050
(512) 338-5423
Client No. 021971